Simulating Selection and Random Genetic Drift

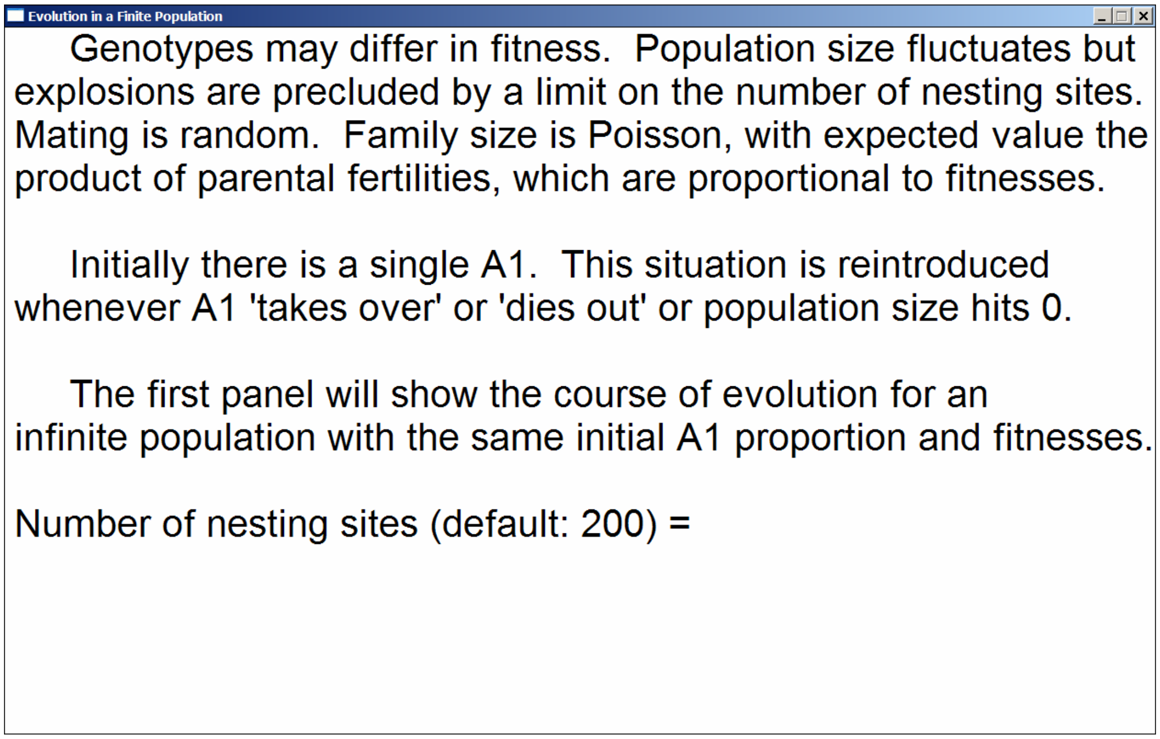
M. Frank Norman (norman@psych.upenn.edu) 8/13/08

The simulator and supporting DLL are in the archive

http://psych.upenn.edu/~norman/EvolSimRandomDrift.zip

There is no installation other than unzipping the DLL into the same folder as the EXE. The simulator ran fine under Windows XP on several computers, though I can't predict how it will run on yours. I have not tested it under Vista.

The startup screen gives some basic information:



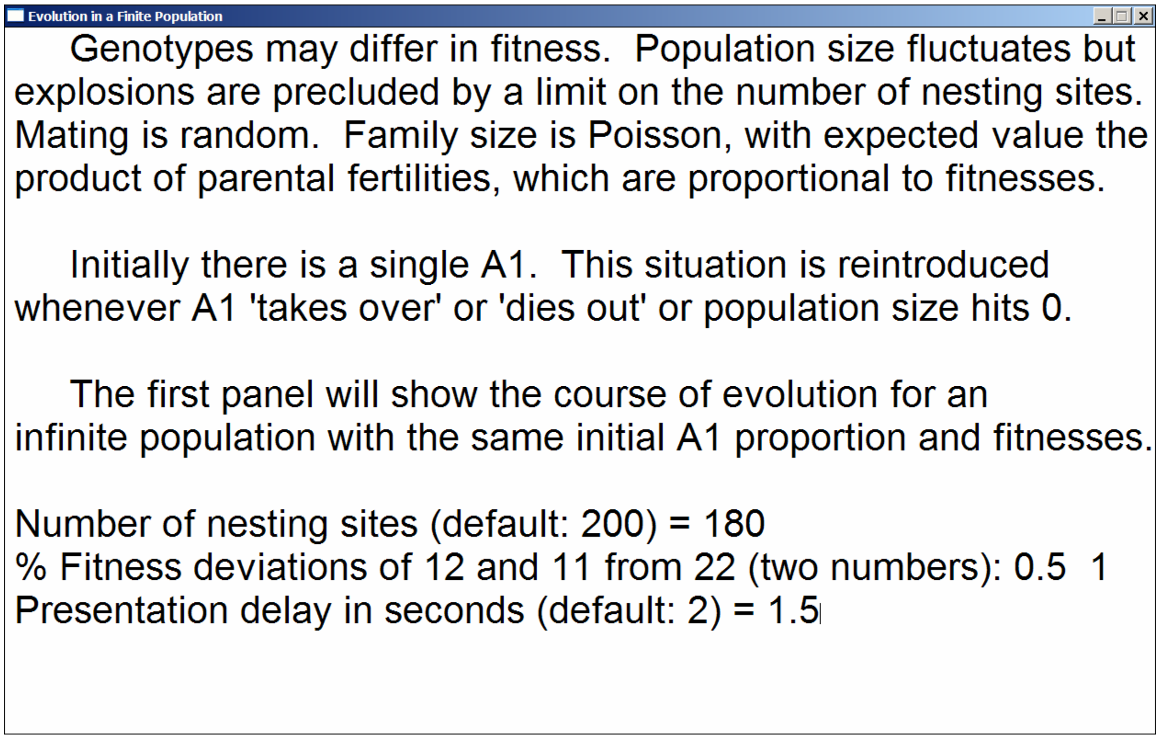
Genotypes may differ in fitness. Population size fluctuates but explosions are precluded by a limit on the number of nesting sites. Mating is random. Family size is Poisson, with expected value the product of parental fertilities, which are proportional to fitnesses.

Initially there is a single A1. This situation is reintroduced whenever A1 'takes over' or 'dies out' or population size hits 0.

The first panel will show the course of evolution for an infinite population with the same initial A1 proportion and fitnesses.

Number of nesting sites (default: 200) =

Selection is controlled by one locus with two alleles, A_1 and A_2. The simulator maintains separate genotype frequencies for males and females. The "number of nesting sites" parameter is the upper limit on the number of randomly mated pairs. The program can certainly handle up to 600 nesting sites, corresponding to a population of approximately 1200. But generations take longer to simulate when the population is larger.



Genotypes may differ in fitness. Population size fluctuates but explosions are precluded by a limit on the number of nesting sites. Mating is random. Family size is Poisson, with expected value the product of parental fertilities, which are proportional to fitnesses.

Initially there is a single A1. This situation is reintroduced whenever A1 'takes over' or 'dies out' or population size hits 0.

The first panel will show the course of evolution for an infinite population with the same initial A1 proportion and fitnesses.

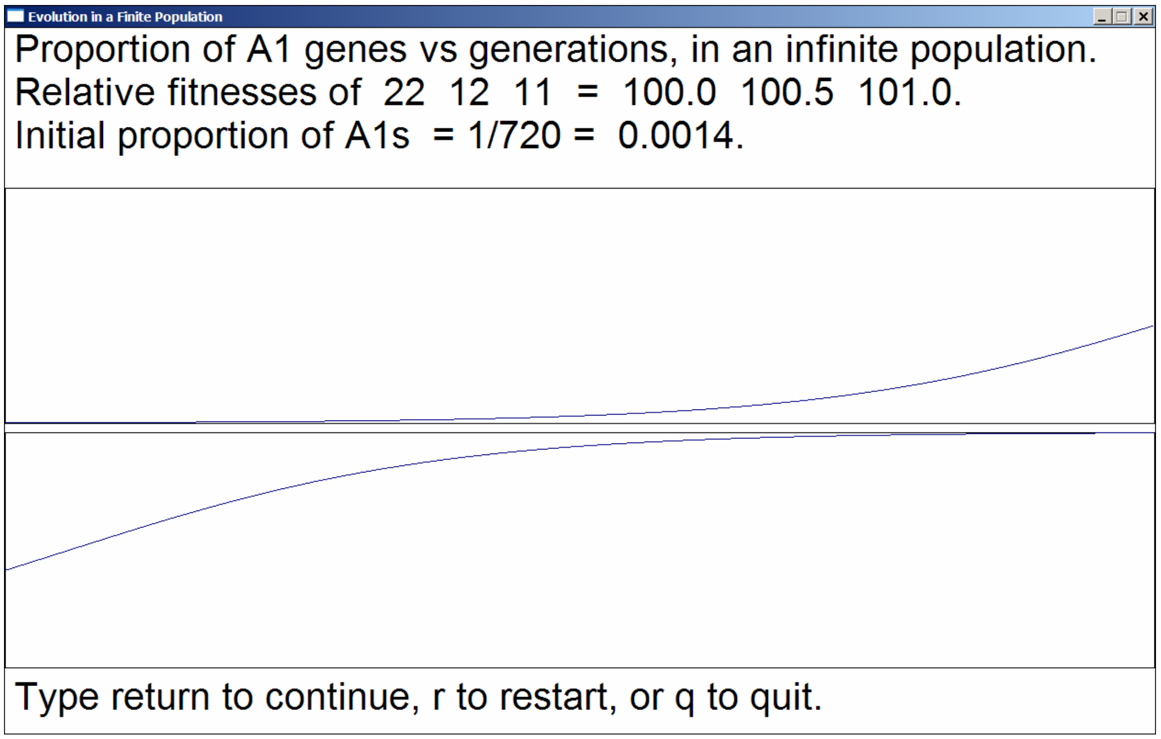
Number of nesting sites (default: 200) = 180
% Fitness deviations of 12 and 11 from 22 (two numbers): 0.5 1
Presentation delay in seconds (default: 2) = 1.5

Here I have specified 180 nesting sites, instead of the default, 200, which would be used if I had pressed Enter without typing anything, and I have specified that relative fitnesses of A_2A_2, A_1A_2, and A_1A_1 are, respectively, 1.0, 1.005 and 1.01. (The default percentage deviations from A_2A_2 fitness are larger, 2% and 4%. Negative entries are permitted.) The program multiplies these relative fitnesses by sqrt(2.5) to obtain genotypic fertility values, which are the same for males and females.

A mated pairs has a random, Poisson-distributed numbers of offspring, with mean the product of the mates' fertility parameters. These products are approximately 2.5, so there are about 2.5 offspring per couple. The 20% excess over 2.0 prevents most population extinctions, and the nesting sites limit prevents population explosions. All newborns survive to reproductive age, so there is no viability-based selection.

The program runs very fast for small populations, filling up and erasing graphic screens too quickly for close examination, so the opening screen permits specification of a delay before erasure. If one is only interested in fast generation of summary statistics, one should specify a 0 delay. And, regardless of the delay specification, output can always be paused and restarted by pressing the Alt key.

After entering the delay, one obtains

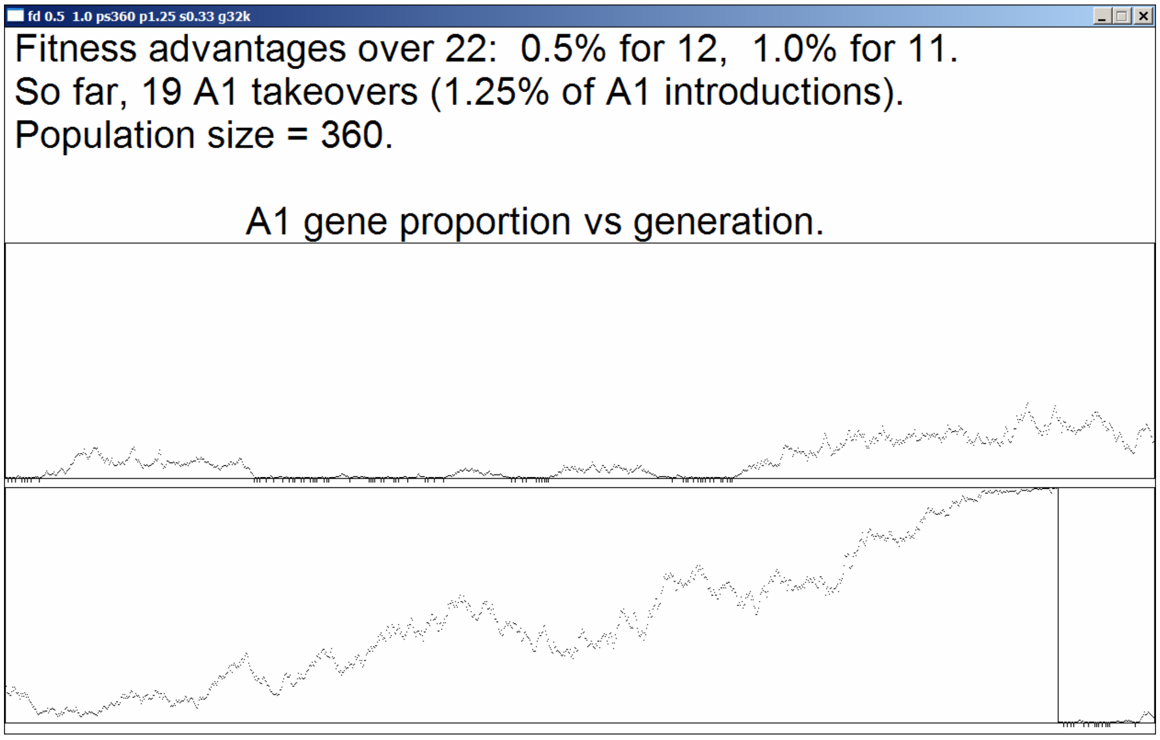


Proportion of A1 genes vs generations, in an infinite population.
Relative fitnesses of 22 12 11 = 100.0 100.5 101.0.
Initial proportion of A1s = 1/720 = 0.0014.

Type return to continue, r to restart, or q to quit.

This screen graphs A_1 gene proportion vs. generations for a traditional, infinite population evolutionary model with the same selection intensity that will be used in our finite population, and with a single initial A_1 gene (e.g., a new mutant or migrant). The population size is approximately twice the number of nesting sites, and each individual has two genes at the locus in question, so the initial proportion of A_1s is the reciprocal of four times the nesting site parameter, 1/720 in the example above. Here and subsequently, the width of the graph is 1000 generations for screens with width 1024 pixels, and 1250 generations for screens with width 1280 or 1920 pixels. The lower panel is (somewhat confusingly) a continuation of the upper panel.

Pressing return starts the actual simulation, which produces a quick succession of output screens like the one below (captured via Alt-PrtSc, and pasted into a Word document).



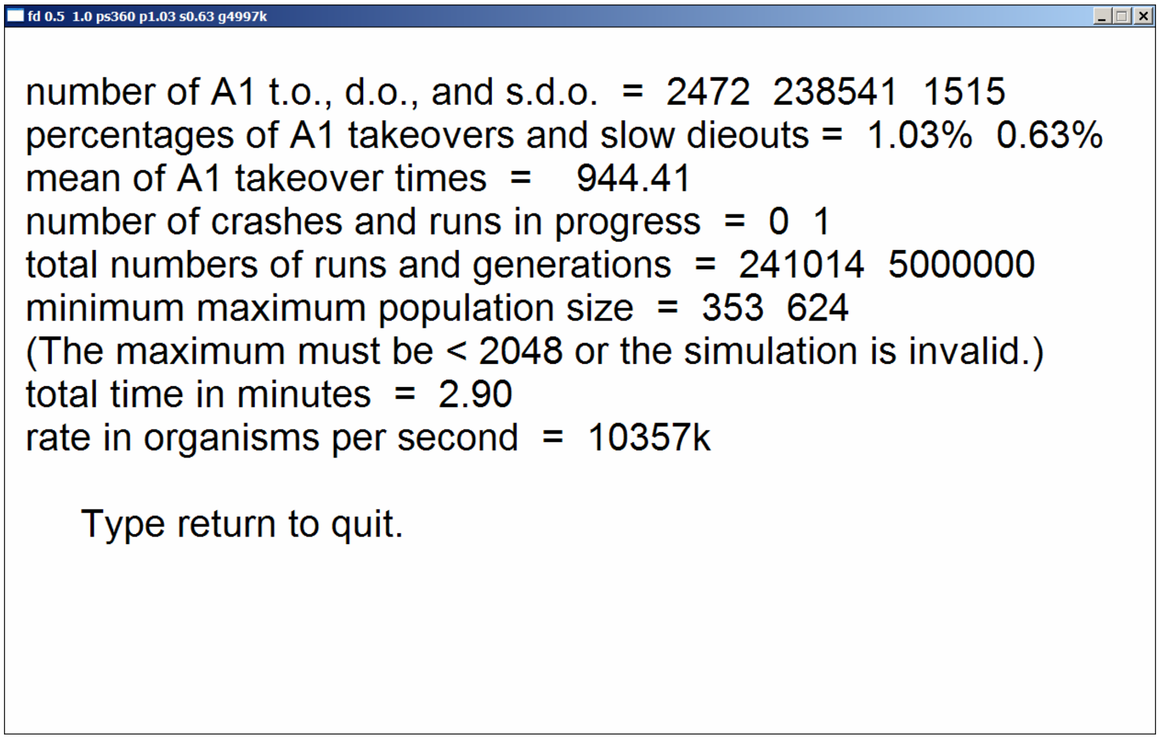
Fitness advantages over 22: 0.5% for 12, 1.0% for 11.
So far, 19 A1 takeovers (1.25% of A1 introductions).
Population size = 360.

A1 gene proportion vs generation.

The simulation consists of repetitions of the same simple scenario. A single A_1 is introduced. After a variable number of generations, A_1 either "takes over," with the A_1 proportion hitting the top of the graph, or (much more likely) "dies out," with A_1 proportion hitting the bottom of the graph. Dieouts are marked by small ticks below the $p = 0$ line, and takeovers are marked by vertical lines from $p = 1$ to $p = 0$. Then the same scenario is repeated.

In the sample screen above, there are a number of quick dieouts, then a dieout that take somewhat longer, then a lot of dieouts, then a very long run (over 1400 generations), extending over both panels, ending in a takeover, then several quick dieouts. The jagged takeover run should be contrasted with the smooth, infinite population graph. Jaggedness is the signature of random drift. At the top of the screen we see that, by this point in the simulation, there have been 19 takeovers, which represent 1.25% of the A_1 introductions. This value is cryptically reproduced in the window title bar ("p1.25"). "g32k" means that about 32,000 generations have transpired. "s0.33" indicates that the proportion of dieouts that are "slow" (requiring more than 200 generations) is 0.33%--most dieouts are very quick.

One can terminate the session at any time by closing the program window, or one can let it run for a full 5 million generations (without inter-screen delays, if time is short), at which time one is rewarded by a full statistical summary like the following:



```
number of A1 t.o., d.o., and s.d.o.  =  2472  238541  1515
percentages of A1 takeovers and slow dieouts =  1.03%  0.63%
mean of A1 takeover times  =    944.41
number of crashes and runs in progress  =  0  1
total numbers of runs and generations  =  241014  5000000
minimum maximum population size  =  353  624
(The maximum must be < 2048 or the simulation is invalid.)
total time in minutes  =  2.90
rate in organisms per second  =  10357k

    Type return to quit.
```

This information is automatically written to a text file in the same directory as the program file. The text file has a numerical name that is chosen so as not to overwrite previous output. It has no extension, so one cannot open it by double clicking.

Note that the program required only 2.90 minutes, or 174 seconds to move a population of size about 360 through 5,000,000 generations. This translates into more than 10,300,000 organisms/second on this relatively fast computer.

I mentioned earlier that the program could handle 600 nesting sites. It can, in fact, handle more. One need only check the output statistics to verify that the maximum population size (after births but before mating) was less than 2048. Remember that each couple has 2.5 offspring on the average, so, absent randomness, there could be almost 2047/2.5 = 818 nesting sites. 700 should work most of the time.

The random number generator is seeded independently each time the program is run, so successive runs with the same parameters will produce different output.

One can run multiple instances of the simulator at the same time, though, of course, all instances run more slowly.